



Hybrid segmentation of depth images using a watershed and region merging based method for tree species recognition

Ahlem Othmani, Alexandre Piboule, Yan Voon

► To cite this version:

Ahlem Othmani, Alexandre Piboule, Yan Voon. Hybrid segmentation of depth images using a watershed and region merging based method for tree species recognition. 11 th IEEE IVMSP workshop : 3D Image / Video Technologies and Applications (IVMSP 2013), Jun 2013, Seoul, South Korea. pp.1 - 4, 10.1109/IVMSPW.2013.6611901 . hal-01116700

HAL Id: hal-01116700

<https://hal.science/hal-01116700>

Submitted on 14 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HYBRID SEGMENTATION OF DEPTH IMAGES USING A WATERSHED AND REGION MERGING BASED METHOD FOR TREE SPECIES RECOGNITION

Ahlem OTHMANI^{1,2}, Alexandre PIBOULE² and Lew F.C. LEW YAN VOON¹

¹Le2i UMR-CNRS 6306, IUT Le Creusot, 12 rue de la fonderie, 71200 Le Creusot, France
e-mail: {ahlem.othmani, lew.lew-yan-voon}@u-bourgogne.fr

²Office National des Forêts, Pôle R&D de Nancy, 11 rue de l'Ile-de-Corse, 54000 Nancy, France
e-mail: alexandre.piboule@onf.fr

ABSTRACT

Tree species recognition from Terrestrial Light Detection and Ranging (T-LiDAR) scanner data is essential for estimating forest inventory attributes in a mixed planting. In this paper, we propose a new method for individual tree species recognition based on the analysis of the 3D geometric texture of tree barks. Our method transforms the 3D point cloud of a 30 cm segment of the tree trunk into a depth image on which a hybrid segmentation method using watershed and region merging techniques is applied in order to reveal bark shape characteristics. Finally, shape and intensity features are calculated on the segmented depth image and used to classify five different tree species using a Random Forest (RF) classifier. Our method has been tested using two datasets acquired in two different French forests with different terrain characteristics. The accuracy and precision rates obtained for both datasets are over 89%.

Index Terms— Forest inventory, terrestrial laser scanning, single tree species recognition, 3D geometric texture classification.

1. INTRODUCTION

The development of software tools for the automatic measurement of forest inventory attributes has become a major research field due to the increasing use of Light Detection and Ranging (LiDAR) technology in the forestry domain [1]. Recently, the National Office of Forests ("Office National des Forêts – ONF") has started to develop a software tool named "Computree" for the automatic measurement of forest inventory attributes using T-LiDAR Scanner (TLS) data [2]. Among the important attributes that "Computree" should be able to determine is the species of each tree in the ground plot. The identification of the species in a mixed planting is essential for estimating forest inventory attributes according to the species and not in a global manner. But, to the best of our knowledge not much has been done regarding single tree species identification based on TLS data. In this paper we present our work on single tree species identification using ground plot TLS

data. The common criteria used to identify the species of a single tree are the shape of the leaves, the general shape of the crown and the variations in geometry across the surface of the bark known as the geometric texture of the bark. Since forest inventory data are mostly acquired during winter in order to reduce occlusions due to leaves and also in order to do the measurement outside the growing period of the trees, it cannot be envisaged to recognize the species based on the shape of the leaves. Finally, the bark is probably the most discriminating feature of the species even if it is subject to changes during the tree's life because of age, injuries and modified growth pattern due to environmental disturbances. Figure 1 shows the bark texture of the five most dominant species in European forest. They are the hornbeam, the oak, the spruce, the beech and the pine.



Figure 1: Example of 3D mesh of the five tree species

One can notice that each of the five species has a distinguishable bark feature: the beech has a relatively smooth surface, the spruce is less smooth compared to the beech and it has circular scars, the pine and the oak are rough with vertical strips but the growth pattern is different, and the hornbeam is smooth with an undulating texture. We thus propose a method that analyzes the geometric variation patterns or geometric texture of the 3D surface of the bark in order to determine the species. The first step of our method, as depicted in section 2, consists thus in generating a 2D depth map representation of the 3D geometric texture from the 3D point cloud of a 30 cm segment of the trunk called a "patch". Then, a hybrid depth image segmentation method is used to extract texture features for tree species classification using the Random Forest technique. Section 3 describes the experimental setting and discusses about the results obtained

on two datasets, D1 and D2, consisting of respectively 16 and 30 patches per species.

2. METHODOLOGY

Our method consists of four steps as represented in Figure 2. Each of the steps, except the first one which is a denoising step using the RapidForm™ software (<http://www.rapidform.com>) for removing ghost points, will be described in the subsequent sections.

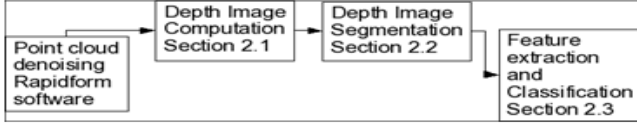


Figure 2: Flow diagram of the method

2.1 Depth image computation

After the denoising step the 3D point cloud of the tree trunk segment is transformed into a 2D depth image that represents the 3D geometric details of the bark. 3D geometric details are defined as the local variations of the original mesh M_o with respect to a smoothed version of the same mesh M_s as represented in Figure 3.

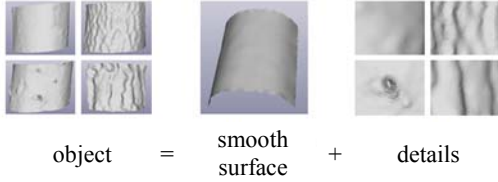


Figure 3: Illustration of 3D geometric details

Thus, the 3D geometric details is a dataset (x, y, z, d) in the \mathbb{R}^4 space where (x, y, z) are the coordinates of the 3D points or vertices of the smoothed mesh and d , known as the details, the Euclidean distance between a point of the smoothed mesh M_s of coordinates (x, y, z) , denoted by $v(x, y, z)$, and its nearest neighbor, denoted by $\tilde{v}(x', y', z')$, in the original mesh M_o .

$$d(v, \tilde{v}) = \sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2} \quad (1)$$

where

$$\tilde{v} = \arg \min_{p_i \in M_o} \|p_i - v\| \quad (2)$$

$\tilde{v}(x', y', z')$ is determined using the efficient Aligned Axis Bounding Box (AABB) tree structure [3].

The smoothed mesh is computed using Taubin's λ/μ smoothing algorithm [4]. It consists in basically performing the Laplacian smoothing two consecutive times with different scaling factors noted λ and μ . A first step with $\lambda > 0$ (shrinking step) and a second step with a negative scaling factor $\mu < -\lambda < 0$ (unshrinking step). Laplacian smoothing consists in iteratively moving each of the vertices of the mesh to a new position that corresponds to the weighted average position of the neighboring vertices. The new position v'_i of a vertex i is given by:

$$v'_i = v_i + \lambda \Delta v_i \quad (3)$$

where v_i is the current position, λ is a scalar that controls the diffusion speed and Δv_i the Laplacian operator, a weighted sum of the difference between the current vertex v_i and its neighbors v_j , given by the following equation:

$$\Delta v_i = \sum_{v_j \in i^*} w_{i,j} (v_j - v_i) \quad (4)$$

where i^* is the set of all the neighbors of the vertex v_i . Taubin's λ/μ smoothing algorithm is run with equal weights $w_{i,j}$ for each of the neighbors such that $\sum_j w_{i,j} = 1$ and with λ and μ respectively equal to 0.6307 and -0.6732 (values suggested by Taubin). It is run iteratively until the mesh is sufficiently smoothed. To measure the smoothness of the mesh the maximum curvature is computed for all the points of the mesh and the median of the maximum curvature values considered. An experimental study that we have conducted showed that the mesh is sufficiently smoothed when the difference in the median of the maximum curvature values between two successive runs of Taubin's algorithm is less than or equal to 1%. We thus use this criterion as the smoothing stopping criterion.

The geometric details, a 3D map or a dataset (x, y, z, d) in the \mathbb{R}^4 space, is next transformed into a depth image, a dataset in the (X, Y, I) space that can be modeled as:

$$I : \mathbb{N}^2 \rightarrow \mathbb{R}$$

$$I(X, Y) = \begin{cases} \text{Mean}(d_i) & \text{if } \exists d_i : \varphi_i = (X, Y) \\ \text{Bicubic interpolation} & \text{if } \nexists d_i : \varphi_i = (X, Y) \end{cases} \quad (5)$$

where d_i is the details for a point v_i of coordinates (x_i, y_i, z_i) given by equation (1) and

$$\varphi_i = \varphi(x_i, y_i, z_i) = (X_i, Y_i) \quad (6)$$

$\varphi(\cdot)$ is the Isomap dimensionality reduction algorithm proposed by Tenenbaum et al [5] followed by a quantization step. The latter step is necessary since Isomap transforms nonlinear manifolds in \mathbb{R}^3 to Euclidean manifolds in \mathbb{R}^2 whereas we would like to transform the 3D coordinates of a non linear manifold to the 2D coordinates $(X, Y) \in \mathbb{N}^2$ of the pixels of the depth image. Isomap is particularly suitable in our case since it minimizes the geodesic distortion between points on the 3D surface which results in a minimization of shape distortion when transforming from \mathbb{R}^3 to \mathbb{R}^2 .

In equation (5) the mean of the distances d_i is computed since there may be several 3D points $v(x, y, z)$ that map to the same pixel at coordinates (X, Y) in the depth image. Also, it is possible that no 3D point $v(x, y, z)$ maps to a pixel of coordinates (X, Y) . In that case a bicubic interpolation is done in order to estimate a value for the pixel. Examples of the depth images in the gray level scale $[0, 255]$ for each of the five species to identify are shown in Figure 4.

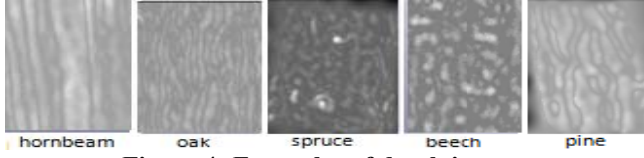


Figure 4: Examples of depth images

2.2 Hybrid edge-region segmentation

The proposed method is a 2-step approach. The first step uses watershed to produce a primary segmentation of the input depth images, while the second step applies a weakness boundary based region-merging algorithm to obtain the final segmented map [6]. The flow diagram of the proposed method is shown in Figure 5.

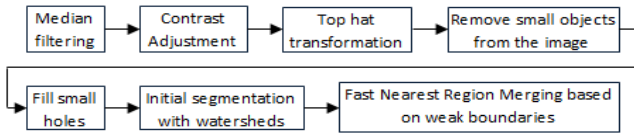


Figure 5: Flow diagram of the segmentation algorithm

2.2.1 Initial segmentation using Watershed

Watershed transformation [6, 7] is a powerful tool for image segmentation. It considers the gray level of a pixel as its altitude in the relief which is exactly the case in depth images. This is why we have chosen this method.

The watershed transform of the gradient magnitude of the image very often leads to an over-segmentation of the image. It gives poor results in the case of the depth images of tree barks even when foreground object and background markers are used because of noises in the depth images. Consequently, we use a function based on a photometric criterion (contrast between facets due to blazing ridges) in the watershed. It is a morphological transformation called “Top-Hat” transformation which is defined by:

$$H = F - (F \circ b) \quad (7)$$

where F is the depth image and \circ is the morphological opening operation with an octagonal structuring element b of size 30 pixels from the origin to the sides of the octagon. It is a contrast detector suitable for enhancing blazing zones in the image. A post processing stage is essential for removing small objects from the image and filling the holes by an opening operation followed by a closing one. The resulting segmented tree trunk depth images are shown in Figure 6.

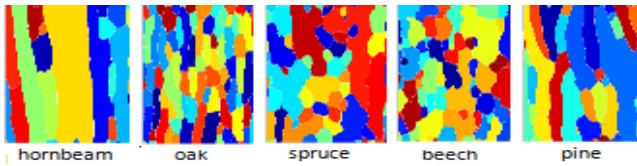


Figure 6: Watershed segmented images

2.2.2 Region merging based on weak boundaries

Despite all the processing in the initial segmentation to overcome the over-segmentation problem, there still remain neighboring regions that could be merged into a more meaningful region. In our case, there can be two over-segmented adjacent regions R_1 and R_2 that are not homogeneous, but, they nevertheless belong to the same real region. An edge dissimilarity measure that evaluates the weakness of the boundaries will be used as the merging criteria. The Region Adjacency Graph (RGA), as illustrated in Figure 7, will be used for the merging. The RAG of a K -partition is defined as an undirected graph, $G = (R, E)$ where $R = \{1, 2, \dots, K\}$ is the set of nodes and $E \subset R \times R$ is the set of edges. Each region is represented by a graph node. Two adjacent regions R_i and R_j are represented by two nodes $i, j \in R$ and connected by the edge (i, j) . The dissimilarity measure between the two regions represents the weight of the graph edge.

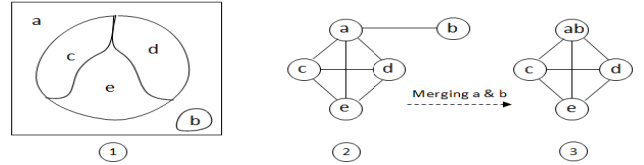


Figure 7: 1) Five-partition of an image, 2) the corresponding RAG, and 3) merging of two nodes.

The region-merging algorithm starts with the initial segmentation. The corresponding edge map in the original image is extracted. Two adjacent regions with an edge cost E_{ij} lower than a fixed threshold (0.5 in our case) are merged.

$$E_{i,j} = \alpha \frac{n_1}{n} + \beta \frac{n_2}{n} \quad (8)$$

where n is the number of pixels in the edge linking the two adjacent regions, n_1 is the number of pixels considered as black one (grayscale < 0.8), n_2 is the number of pixels considered as white pixels (grayscale > 0.8), $\alpha = 0.75$ and $\beta = 0.25$. The final segmentation is shown in Figure 8.

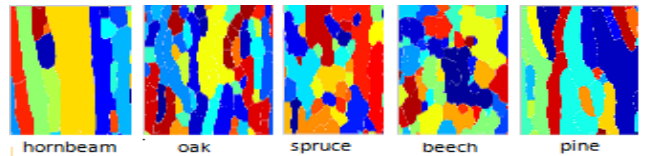


Figure 8: Segmented images after region merging

2.3 Feature extraction and classification

We assign each region a set of values representing quantitative attributes that describes the geometry and the directionality. We used the regionprops() function of Matlab to calculate the properties of regions including area, perimeter, orientation, and so on. In addition to those descriptors, we used other shape features: compactness, circularity ratio, rectangularity, concavity, and convexity, solidity as described in [8].

Severe injury occurs during the life of the tree which can damage the tree bark and change its discriminating criteria. Also, the size of the scars depends strongly on the age of the tree and the distance to the scanner which can result in a dispersion of the features. To evaluate the intra-class variability, we consider the mean, the standard deviation and the median of each feature of all the regions. Classification is done using the Random Forest (RF) classifier proposed by Breiman [9] in order to classify the textures and thus the tree species.

3. EXPERIMENTAL RESULTS AND CONCLUSION

We used two different datasets D1 and D2 to validate our approach experimentally. The test site of D1 is a mixed state forest of Montiers-sur-Saulx, France. The second test site of D2 is a mixed forest stands of grove and coppice-under-grove forests in Lorraine, France. The T-LiDAR data was collected with a Faro photon 120™ scanner. A segmentation of the 3D plots as described in [2] is done and 3D patches are extracted. The two datasets contain five different species of trees (Figure 1): 16 patches per species in D1 and 30 patches per species in D2. Two tests are done and in each test we train a separate RF classifier built with 1000 decisions trees. For both D1 and D2, we split the dataset into 10 equal size subsets and performed a 10-fold cross-validation. The confusion matrices are reported in Tables 1 and 2.

	1	2	3	4	5
1	14.428	0.571	0	0	1
2	0.428	12.571	1.714	0	1.285
3	0.285	0	15.142	0.428	0.142
4	0	0	0.714	15.142	0.142
5	0.714	0.142	0.857	0	14.285

Table 1: Confusion matrix for D1 cross-validation (1: hornbeam, 2: oak, 3: spruce, 4: beech, 5: pine)

	1	2	3	4	5
1	27.166	0.166	0	1.833	0.833
2	0	29	0	0	1
3	0.166	1.166	27.333	1.333	0
4	1	0	1.333	27.333	0.333
5	0	0	0.166	1.5	28.333

Table 2: Confusion matrix for D2 cross-validation

For both datasets, each with different terrains and architecture characteristics, we obtain good accuracy and precision rates. The achieved overall accuracy and precision in tree species recognition are respectively 95% and 89% for

D1 and 97% and 92% for D2. We can differentiate two families from the five tested species (species that have straps or cracks: hornbeam, oak and pine) and the family of random shapes (smooth species as beech and spruce). We note from the confusion matrices that our algorithm mainly mixes intra-family species but also inter-family species.

Future work involves finding better features to discriminate inter and intra family species and the validation of the performance of our method when changing the distance to the scanner.

4. ACKNOWLEDGEMENT

This work is financially supported by the "Conseil Regional de Bourgogne" under contract N° 2010-9201AAO048S06469 and N° 2010-9201CPERO007S06470, and the "Office National des Forêts".

5. REFERENCES

- [1] M. Dassot, T. Constant and M. Fournier, "The Use of Terrestrial LiDAR Technology in Forest Science: Application fields, Benefits and Challenges", *Annals of Forest Science*, Vol. 6, pp. 959-974, 2011.
- [2] A. Othmani, A. Piboule, M. Krebs, C. Stolz and L.F.C. Lew Yan Voon, "Towards Automated and Operational Forest Inventories with T-LiDAR", *SilviLaser*, 2011.
- [3] P. Alliez, S. Tayeb and C. Wormser, "AABB Tree", *CGAL 3.5 edition*, 2009.
- [4] G. Taubin, "Geometric Signal Processing on Polygonal Meshes", *State of the Art Report*, Eurographics, 2000.
- [5] J.B. Tenenbaum, Vin de Silva and J.C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction", *Science*, pp. 2319-2323, December 2000.
- [6] K.Haris, N. Efstratiadis, N. Maglaveras and A.K. Katsaggelos, "Hybrid Image Segmentation Using Watersheds and Fast Region Merging", *IEEE Trans Image Processing*, pp. 1684-1699, 1998.
- [7] S. Beucher and C. Lantuéjoul, "Use of Watersheds in Contour Detection", *International workshop on image processing, real-time edge and motion detection*, 1979.
- [8] Y. Mingqiang, K. Kidiyo and R. Joseph, "A Survey of Shape Feature Extraction Techniques", *Pattern Recognition*, pp. 43-90, Nov. 2008.
- [9] L. Breiman, "Random Forests", *Machine Learning*, pp. 5-32, October 2001.